Apple Inc.

Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1]
FIPS 140-3 Non-Proprietary Security Policy

October 2024

Prepared for:

Apple

One Apple Park Way

Cupertino, CA 95014

Prepared by:

atsec information security corporation

4516 Seton Center Parkway, Suite 250

Austin, TX 78759

www.atsec.com

Trademarks

Apple's trademarks applicable to this document are listed in https://www.apple.com/legal/intellectual-property/trademark/appletmlist.html.

Other company, product, and service names may be trademarks or service marks of others.

Table of Contents

List of Tables

1 General

This document is the non-proprietary FIPS 140-3 Security Policy for Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1] cryptographic module. It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for a Security Level 1 module.

This document provides all tables and diagrams (when applicable) required by NIST SP 800-140B. The column names of the tables follow the template tables provided in NIST SP 800-140B.

Table 1 describes the individual security areas of FIPS 140-3, as well as the Security Levels of those individual areas.

ISO/IEC 24759 Section 6.[Number Below]	FIPS 140-3 Section Title	Security Level
1	General	1
2	Cryptographic Module Specification	1
3	Cryptographic Module Interfaces	1
4	Roles, Services, and Authentication	1
5	Software/Firmware Security	1
6	Operational Environment	1
7	Physical Security	Not Applicable
8	Non-invasive Security	Not Applicable
9	Sensitive Security Parameter Management	1
10	Self-tests	1
11	Life-cycle Assurance	1
12	Mitigation of Other Attacks	Not Applicable

Table 1 - Security Levels

2 Cryptographic Module Specification

The Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1] cryptographic module (hereafter referred to as “the module”) is a Software module running on a multi-chip standalone general-purpose computing platform. The version of module is 12.0. The module provides implementations of low-level cryptographic primitives to the Device OS’s (iOS 15, iPadOS 15, watchOS 8, tvOS 15, T2 OS 12 and macOS 12 Monterey) Security Framework and Common Crypto.

2.1 Tested Operational Environments

#	Operating System	Hardware Platform	Processor	PAA/Acceleration
1	iPadOS 15	iPad (5th generation)	Apple A Series A9	With and without PAA
2	iPadOS 15	iPad Pro 9.7-inch	Apple A Series A9X	With and without PAA
3	iPadOS 15	iPad (7th generation)	Apple A Series A10 Fusion	With and without PAA
4	iPadOS 15	iPad Pro 10.5 inch	Apple A Series A10X Fusion	With and without PAA
5	iPadOS 15	iPad mini (5th generation)	Apple A Series A12 Bionic	With and without PAA
6	iPadOS 15	iPad Pro 11-inch (1st generation)	Apple A Series A12X Bionic	With and without PAA
7	iPadOS 15	iPad Pro 11-inch (2nd generation)	Apple A Series A12Z Bionic	With and without PAA
8	iPadOS 15	iPad (9th generation)	Apple A Series A13 Bionic	With and without PAA
9	iPadOS 15	iPad Air (4th generation)	Apple A Series A14 Bionic	With and without PAA
10	iPadOS 15	iPad mini (6th generation)	Apple A Series A15 Bionic	With and without PAA
11	iPadOS 15	iPad Pro 11-inch (3rd generation)	Apple M Series M1	With and without PAA
12	iOS 15	iPhone 6S	Apple A Series A9	With and without PAA
13	iOS 15	iPhone 7 Plus	Apple A Series A10 Fusion	With and without PAA
14	iOS 15	iPhone X	Apple A Series A11 Bionic	With and without PAA
15	iOS 15	iPhone XS Max	Apple A Series A12 Bionic	With and without PAA
16	iOS 15	iPhone 11 Pro	Apple A Series A13 Bionic	With and without PAA
17	iOS 15	iPhone 12	Apple A Series A14 Bionic	With and without PAA
18	iOS 15	iPhone 13 Pro Max	Apple A Series A15 Bionic	With and without PAA
19	watchOS 8	Apple Watch Series S3	Apple S Series S3	With and without PAA
20	watchOS 8	Apple Watch Series S4	Apple S Series S4	With and without PAA
21	watchOS 8	Apple Watch Series S5	Apple S Series S5	With and without PAA
22	watchOS 8	Apple Watch Series S6	Apple S Series S6	With and without PAA
23	watchOS 8	Apple Watch Series S7	Apple S Series S7	With and without PAA
24	tvOS 15	Apple TV 4K	Apple A Series A10X Fusion	With and without PAA
25	tvOS 15	Apple TV 4K (2nd generation)	Apple A Series A12 Bionic	With and without PAA
26	T2OS 12	Apple Security Chip T2	Apple T Series T2	With and without PAA
27	macOS 12 Monterey	MacBook Pro (13-inch, M1, 2020)	Apple M Series M1	With and without PAA
28	macOS 12 Monterey	MacBook Pro 14-inch	Apple M Series M1 Pro	With and without PAA
29	macOS 12 Monterey	MacBook Pro 16-inch	Apple M Series M1 Max	With and without PAA

Table 2 - Tested Operational Environments

2.2 Vendor-affirmed Operational Environments

#	Operating System	Hardware Platform
1	iPadOS 15	iPad Pro 12.9-inch
2	iPadOS 15	iPad (6th generation)
3	iPadOS 15	iPad Pro 12.9-inch (2nd generation)
4	iPadOS 15	iPad Air (3rd generation)
5	iPadOS 15	iPad (8th generation)

#	Operating System	Hardware Platform
6	iPadOS 15	iPad Pro 12.9-inch (3rd generation)
7	iPadOS 15	iPad Pro 12.9-inch (4th generation)
8	iPadOS 15	iPad Pro 12.9-inch (5th generation)
9	iOS 15	iPhone SE
10	iOS 15	iPhone 6S Plus
11	iOS 15	iPhone 7
12	iOS 15	iPhone 8
13	iOS 15	iPhone 8 Plus
14	iOS 15	iPhone XS
15	iOS 15	iPhone XR
16	iOS 15	iPhone 11
17	iOS 15	iPhone 11 Pro Max
18	iOS 15	iPhone SE (2nd generation)
19	iOS 15	iPhone 12 mini
20	iOS 15	iPhone 12 Pro
21	iOS 15	iPhone 12 Pro Max
22	iOS 15	iPhone 13 mini
23	iOS 15	iPhone 13
24	iOS 15	iPhone 13 Pro
25	watchOS 8	Apple Watch SE
26	macOS 12 Monterey	MacBook Air
27	macOS 12 Monterey	Mac mini
28	macOS 12 Monterey	iMac (24-inch)

Table 3 - Vendor Affirmed Operational Environments

The CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

2.3 Modes of operation

The module operates in Approved and Non-Approved mode of operation. The mode is implicit and is based on the service utilized. The table below provides a summary of the implementation.

Name	Description	Type	Status Indicator
Approved mode	Approved mode of operation is entered when the module utilizes the services that use the security functions listed in the Table 5 and Table 6.	Approved mode	return a '1' from fips_allowed_mode() for block cipher functions and fips_allowed() for all other services to indicate the executed cryptographic algorithm was approved
Non-Approved mode	Non-Approved mode of operation is entered when the module utilizes non-approved security functions in Table 7.	Non-Approved mode	return a '0' from fips_allowed_mode() for block cipher functions and fips_allowed() for all other services to indicate the executed cryptographic algorithm was non-approved

Table 4 - Modes of Operation

2.4 Vendor Affirmed Algorithms

Algorithm	Algorithm Properties	Use / Function
CKG [SP800-133r2] (asymmetric)	Vendor affirmed	Cryptographic key Generation for ECDSA key pair; FIPS 140-3 IG D.H and SP800-133r2 section 4 example 1

Table 5 - Vendor Affirmed Algorithms

2.5 Approved Algorithms

The table below lists all Approved or Vendor-affirmed security functions of the module, including specific key size(s) employed for approved services, and implemented modes of operation.

CAVP Cert.	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2793, A2865 (asm_arm)	AES [FIPS 197] [SP 800-38A]	CBC	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	AES [FIPS 197] [SP 800-38A]	CBC	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2796, A2868 (vng_asm)	AES [FIPS 197] [SP 800-38A] [SP 800-38C] [SP 800-38D]	CCM	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2793, A2865 (asm_arm)	AES [FIPS 197] [SP 800-38A]	CFB128	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	AES [FIPS 197] [SP 800-38A]	CFB128	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	AES [FIPS 197] [SP 800-38A]	CFB8	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	AES [FIPS 197] [SP 800-38A]	CTR	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2796, A2868 (vng_asm)	AES [FIPS 197] [SP 800-38A] [SP 800-38C] [SP 800-38D]	CTR	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2793, A2865 (asm_arm)	AES [FIPS 197] [SP 800-38A]	ECB	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	AES [FIPS 197] [SP 800-38A]	ECB	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2796, A2868 (vng_asm)	AES [FIPS 197] [SP 800-38A]	ECB	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption

CAVP Cert.	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2796, A2868 (vng_asm)	AES [FIPS 197] [SP 800-38A] [SP 800-38C] [SP 800-38D]	GCM	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2793, A2865 (asm_arm)	AES [FIPS 197] [SP 800-38A]	OFB	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	AES [FIPS 197] [SP 800-38A]	OFB	Key Size / Key Strength: 128, 192, 256 bits	Symmetric Encryption and Decryption
A2793, A2865 (asm_arm)	AES [FIPS 197] [SP 800-38E]	XTS	Key Size / Key Strength: 128, 256 bits	Symmetric Encryption and Decryption
A2794, A2866 (c_asm)	KTS (AES) [SP 800-38F]	AES-KW	Key Size / Key Strength: 128, 192, 256 bits	Key Wrapping
A2794, A2866 (c_asm)	DRBG [SP800-90ARev1]	CTR_DRBG: AES-128, AES-256	Key Size / Key Strength: 128, 256 bits Derivation Function Enabled, No Prediction Resistance	Random Number Generation
A2796, A2868 (vng_asm)	DRBG [SP800-90ARev1]	CTR_DRBG: AES-128, AES-256	Key Size / Key Strength: 128, 256 bits Derivation Function Enabled, No Prediction Resistance	Random Number Generation
A2797, A2869 (vng_ltc)	RSA [FIPS 186-4]	PKCS#1 v1.5 and PKCS PSS	Key Size: 2048, 3072, 4096 bits Key Strength: from 112 to 150 bits	Digital Signature Generation
A2797, A2869 (vng_ltc)	RSA [FIPS 186-4]	PKCS#1 v1.5 and PKCS PSS	Key Size: 1024 (legacy), 2048, 3072, 4096 bits Key Strength: from 80 to 150 bits	Digital Signature Verification
A2797, A2869 (vng_ltc)	ECDSA ANSI X9.62 [FIPS 186-4]	Key Pair Generation (CKG) using method in Section 4 example 1 of SP 800-133r2. [FIPS 186-4] Appendix B.4.2 Testing Candidates	Curve: P-224, P-256, P-384, P-521 Key Strength: from 112 to 256 bits	Asymmetric Key Generation
A2797, A2869 (vng_ltc)	ECDSA ANSI X9.62 [FIPS 186-4]	N/A	Curve: P-224, P-256, P-384, P-521 bits Key Strength: from 112 to 256 bits	Asymmetric Key Validation
A2797, A2869 (vng_ltc)	ECDSA ANSI X9.62 [FIPS 186-4]	SHA2-224, SHA2-256, SHA2-384, SHA2-512	Curve: P-224, P-256, P-384, P-521 bits Key Strength: from 112 to 256 bits	Digital Signature Generation
A2797, A2869 (vng_ltc)	ECDSA ANSI X9.62 [FIPS 186-4]	SHA1 (legacy), SHA2-224, SHA2-256, SHA2-384, SHA2-512	Curve: P-224, P-256, P-384, P-521 bits Key Strength: from 112 to 256 bits	Digital Signature Verification
A2797, A2869 (vng_ltc)	SHS [FIPS 180-4]	SHA-1	N/A	Message Digest

CAVP Cert.	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2797, A2869 (vng_ltc)	SHS [FIPS 180-4]	SHA-224	N/A	Message Digest
A2797, A2869 (vng_ltc)	SHS [FIPS 180-4]	SHA-256	N/A	Message Digest
A2797, A2869 (vng_ltc)	SHS [FIPS 180-4]	SHA-384	N/A	Message Digest
A2797, A2869 (vng_ltc)	SHS [FIPS 180-4]	SHA-512	N/A	Message Digest
A2797, A2869 (vng_ltc)	SHS [FIPS 180-4]	SHA-512/256	N/A	Message Digest
A2795, A2867 (c_ltc)	SHS [FIPS 180-4]	SHA-384	N/A	Message Digest
A2795, A2867 (c_ltc)	SHS [FIPS 180-4]	SHA-512	N/A	Message Digest
A2795, A2867 (c_ltc)	SHS [FIPS 180-4]	SHA-512/256	N/A	Message Digest
A2798, A2870 (vng_neon)	SHS [FIPS 180-4]	SHA-256 for all CPUs in Table 2 except S3)	N/A	Message Digest
A2797, A2869 (vng_ltc)	HMAC [FIPS 198]	SHA-1	Key Size: 128 - 262144 bits Key Strength: 128 bits	Message authentication (MAC)
A2797, A2869 (vng_ltc)	HMAC [FIPS 198]	SHA-224	Key Size: 224 - 262144 bits Key Strength: 224 bits	Message authentication (MAC)
A2797, A2869 (vng_ltc)	HMAC [FIPS 198]	SHA-256	Key Size: 256 - 262144 bits Key Strength: 256 bits	Message authentication (MAC)
A2797, A2869 (vng_ltc)	HMAC [FIPS 198]	SHA-384	Key Size: 384 - 262144 bits Key Strength: 384 bits	Message authentication (MAC)
A2797, A2869 (vng_ltc)	HMAC [FIPS 198]	SHA-512	Key Size: 512 - 262144 bits Key Strength: 512 bits	Message authentication (MAC)
A2797, A2869 (vng_ltc)	HMAC [FIPS 198]	SHA-512/256	Key Size: 512 - 262144 bits Key Strength: 256 bits	Message authentication (MAC)
A2795, A2867 (c_ltc)	HMAC [FIPS 198]	SHA-384	Key Size: 384 - 262144 bits Key Strength: 384 bits	Message authentication (MAC)
A2795, A2867 (c_ltc)	HMAC [FIPS 198]	SHA-512	Key Size: 512 - 262144 bits Key Strength: 512 bits	Message authentication (MAC)
A2795, A2867 (c_ltc)	HMAC [FIPS 198]	SHA-512/256	Key Size: 512 - 262144 bits Key Strength: 256 bits	Message authentication (MAC)
A2798, A2870 (vng_neon)	HMAC [FIPS 198]	SHA-256 (for all CPUs in Table 2 except S3)	Key Size: 256 - 262144 bits Key Strength: 256 bits	Message authentication (MAC)

Table 6 - Approved Algorithms

2.6 Non-Approved Algorithms Allowed in the Approved Mode of Operation

There are no non-Approved but "Allowed functions" with security claimed algorithms in approved mode.

2.7 Non-Approved Algorithms Allowed in the Approved Mode of Operation with No Security Claimed

There are no non-Approved Allowed functions with no security claimed algorithms in approved mode.

2.8 Non-Approved Algorithms Not Allowed in the Approved Mode of Operation

The table below lists the non-Approved algorithms and security functions that are used in the non-Approved mode of operation:

Algorithm/Functions	Use / Function
RSA Signature Generation	PKCS#1 v1.5 and PSS Signature Generation Key Size < 2048
RSA Signature Verification	PKCS#1 v1.5 and PSS Signature Verification Key Size < 1024
RSA Key Wrapping	OAEP, PKCS#1 v1.5 and -PSS schemes
Ed25519	Key Agreement Key Generation Signature Generation Signature Verification
ANSI X9.63 KDF	Hash based Key Derivation Function
RFC6637	Key Derivation Function
HKDF [SP800-56C]	Key Derivation Function
DES	Encryption / Decryption Key Size 56-bits
CAST5	Encryption / Decryption Key Sizes 40 to 128-bits in 8-bit increments
AES-GCM using external IV	Authenticated Encryption / Decryption
RC4	Encryption / Decryption Key Sizes 8 to 4096-bits
RC2	Encryption / Decryption Key Sizes 8 to 1024-bits
MD2	Message Digest Digest size 128-bit
MD4	Message Digest Digest size 128-bit
MD5	Message Digest Digest size 128-bit
RIPEMD	Message Digest Digest size 160-bits
ECDSA	PKG: Curve P-192 with security strength of 96 bits PKV: Curve P-192 Signature Generation: Curve P-192 Signature Verification: Curve P-192
	Key Pair Generation for compact point representation of points
Integrated Encryption Scheme on elliptic curves (ECIES)	Hybrid Encryption scheme
Blowfish	Encryption / Decryption
OMAC (One-Key CBC MAC)	MAC generation / verification

Algorithm/Functions	Use / Function
Triple-DES [SP 800-67]	Encryption/Decryption with modes CBC, ECB

Table 7 - Non-Approved Algorithms Not Allowed in the Approved Mode of Operation

2.9 Module components

Package/File Names	Software Version	Integrity Test Implemented
corecrypto-1217.40.11	12.0	HMAC-SHA-256

Table 8 - Executable Code Sets

The module cryptographic boundary is delineated by the dotted green rectangle in the Figure 1. The Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1] executes within the kernel space of the computing platforms and operating systems listed in Table 2 - Tested Operational Environments. In the block diagram below, the Kernel Extension (KEXT) is a bundle that performs low-level tasks. KEXTs run in kernel space, which gives them elevated privileges and the ability to perform tasks that user-space apps can't.

The tested operational environment's physical perimeter (TOEPP) is represented by the most exterior black line in the block diagram Figure 1.

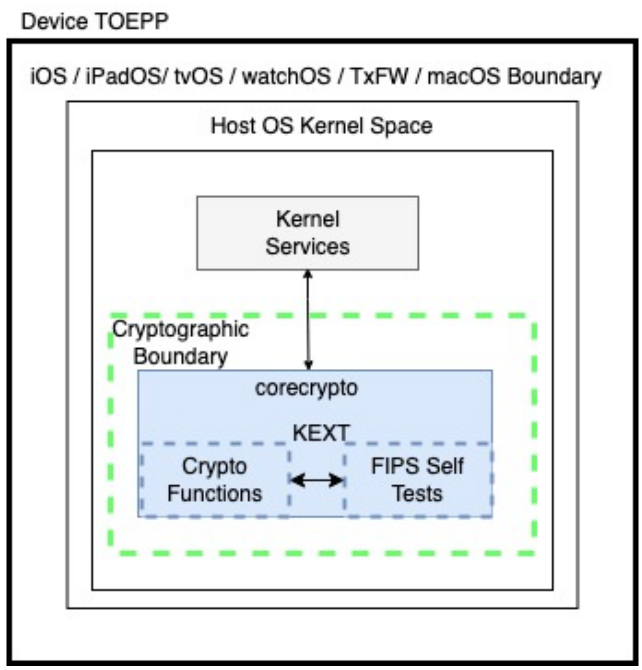

Figure 1 - Block diagram

3 Cryptographic Module Interfaces

The underlying logical interfaces of the module are the C language Kernel Interfaces (KPIs). In detail these interfaces are described in (Table 9):

Physical Ports	Logical Interface1	Data that passes over port/interface
As a software-only module, the module does not have physical ports. Physical Ports are interpreted to be the physical ports of the hardware platform on which it runs	Data Input	Data inputs are provided in the variables passed in the KPI and callable service invocations, generally through caller-supplied buffers
	Data Output	Data outputs are provided in the variables passed in the KPI and callable service invocations, generally through caller-supplied buffers
	Control Input	Control inputs which control the mode of the module are provided through dedicated parameters, namely the kernel module plist whose information is supplied to the module by the kernel module loader.
	Status Output	Status output is provided in return codes and through messages. Documentation for each KPI lists possible return codes. A complete list of all return codes returned by the C language KPIs within the module is provided in the header files and the KPI documentation. Messages are also documented in the KPI documentation.

Table 9 - Ports and Interfaces

The module is optimized for library use within the Device OS kernel space and does not contain any terminating assertions or exceptions. It is implemented as a Device OS dynamically loadable library. The dynamically loadable library is loaded into the Device OS kernel and its cryptographic functions are made available to Device OS kernel services only. Any internal error detected by the module is returned to the caller with an appropriate return code. The calling Device OS kernel service must examine the return code and act accordingly.

The module communicates any error status synchronously through the use of its documented return codes, thus indicating the module's status.

Caller-induced or internal errors do not reveal any sensitive material to callers.

[1] The module does not implement a Control Output Logical Interface

4 Roles, services, and authentication

4.1 Roles

The module supports a single instance of one authorized role: The Crypto Officer. No support is provided for multiple concurrent operators.

Name	Type	Operator Type	Authentication Method
Crypto Officer	Role	CO	Implicit

Table 10 - Roles

4.2 Authentication

FIPS 140-3 does not require an authentication mechanism for level 1 modules. Therefore, the module does not support an authentication mechanism for Crypto Officer. The Crypto Officer role is authorized to access all services provided by the module (see Table 12 - Approved Services and Table 13 - Non-Approved Services below).

4.3 Services

Name	Type	Description	SF Properties	Algorithm Properties
KTS	KTS	SP 800-38F, IG D.G. AES Key Wrapping and Unwrapping	128, 192, and 256-bit AES keys providing 128, 192, or 256 bits of encryption strength	AES-KW/ A2794, A2866

Table 11 - Security Function Implementations

The module implements a dedicated KPI function to indicate if a requested service utilizes an approved security function. For services listed in Table 12 - Approved Services, the indicator function returns 1 to indicate that the security function is approved.

Name	Description	Indicator	Inputs	Outputs	Approved Security Functions	Roles	Access rights to Keys and/ or SSPs
Symmetric Encryption	Executes AES-mode encrypt operation	1	AES key, plaintext data	ciphertext data	AES-CBC, AES-CCM, AES-CFB128, AES-CFB8, AES-CTR, AES-ECB, AES-GCM, AES-OFB, AES-XTS	CO	W, E
Symmetric Decryption	Executes AES-mode decrypt operation	1	AES key, ciphertext data	plaintext data	AES-CBC, AES-CCM, AES-CFB128, AES-CFB8, AES-CTR, AES-ECB, AES-GCM, AES-OFB, AES-XTS	CO	W, E

Name	Description	Indicator	Inputs	Outputs	Approved Security Functions	Roles	Access rights to Keys and/ or SSPs
AES Key Wrapping	Executes AES-key wrapping operation	1	AES key wrapping key, unwrapped key	wrapped key	AES-KW	CO	W, E
AES Key Unwrapping	Executes AES-key unwrapping operation	1	AES key wrapping key, wrapped key	unwrapped key	AES-KW	CO	W, E
Message Digest Generation	Generate a digest for the requested algorithm	1	Message	message digest	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256	CO	N/A
Message Authentication Code (MAC) Generation	Generate a Message Authentication Code	1	HMAC key, MAC algorithm, message	MAC	HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, HMAC-SHA-512/256	CO	W, E
Signature generation (RSA)	Sign a message with a specified RSA private key	1	RSA private key, message, hash algorithm	computed signature	RSA SigGen	CO	W, E
Signature verification (RSA)	Verify the signature of a message with a specified RSA public key	1	RSA public key, digital signature, message, hash algorithm	pass/fail result of digital signature verification	RSA SigVer	CO	W, E
Signature generation (ECDSA)	Sign a message with a specified ECDSA private key	1	ECDSA private key, message, hash algorithm	computed signature	ECDSA SigGen	CO	W, E
Signature verification (ECDSA)	Verify the signature of a message with a specified ECDSA public key	1	ECDSA public key, digital signature, message, hash algorithm	pass/fail result of digital signature verification	ECDSA SigVer	CO	W, E
Random number generation	Generate Random number	1	Output length	Random bit-string	CTR_DRBG (Entropy Input, DRBG seed, Internal state V value and key)	CO	E/ G, W, E / G, W, E
key pair generation (ECDSA)	Generate a keypair for a requested elliptic curve	1	key size	ECDSA Key Pair	ECDSA KeyGen, CKG	CO	G, R
Public key validation (ECDSA)	Verify a public key for a requested elliptic curve	1	ECDSA public key	pass/fail result of key pair verification	ECDSA KeyVer	CO	E, W
Zeroization	Release all resources of symmetric crypto function context	1	N/A	N/A	N/A	CO	Z

Name	Description	Indicator	Inputs	Outputs	Approved Security Functions	Roles	Access rights to Keys and/ or SSPs
	Release all resources of hash context	1	N/A	N/A	N/A	CO	Z
	Release of all resources of asymmetric crypto function context	1	N/A	N/A	N/A	CO	Z
Self-test	Execute the CASTs	1	None	pass/fail results of self-tests	Algorithms listed in table Conditional self-test	CO	N/A
Show Status	Return the module status	None	None	status output	N/A	CO	N/A
Show Module Info	Return Module Base Name and Module Version Number	None	None	name and version information	N/A	CO	N/A

Table 12 - Approved Services

The abbreviations of the access rights to keys and SSPs have the following interpretation:

G = **Generate**: The module generates or derives the SSP.

R = **Read**: The SSP is read from the module (e.g., the SSP is output).

W = **Write**: The SSP is updated, imported, or written to the module.

E = **Execute**: The module uses the SSP in performing a cryptographic operation.

Z = **Zeroise**: The module zeroises the SSP.

N/A= The service does not access any SSP during its operation

Service	Description	Algorithms Accessed	Role
Triple-DES encryption / decryption	TDES-CBC, TDES-ECB	Triple-DES	CO
RSA Key Wrapping	The CAST does not perform the full KTS, only the raw RSA encrypt/ decrypt.	RSA encrypt/decrypt	CO
RSA Signature Generation	PKCS#1 v1.5 and PSS Signature Generation Key Size < 2048	RSA Signature Generation	CO
RSA Signature Verification	PKCS#1 v1.5 and PSS Signature Verification Key Size < 1024	RSA Signature Verification	CO
ECDSA Key-pair Generation (PKG) and ECDSA Key Validation (PKV)	ECDSA PKG and PKV using curve P-192	ECDSA Key Generation, ECDSA Key Validation	CO
ECDSA Signature Generation	ECDSA Signature Generation using curve P-192	ECDSA Signature Generation	CO
ECDSA Signature Verification	ECDSA Signature Verification using curve P-192	ECDSA Signature Verification	CO

Service	Description	Algorithms Accessed	Role
ECDSA Key Pair Generation for compact point representation of points	Key Pair Generation for compact point representation of points	ECDSA Key Generation	CO
Ed25519 Key Generation	Ed25519 Key Generation	Ed25519 Key Generation	CO
Ed25519 Signature Generation	EdDSA Signature Generation over Curve25519	Ed25519 Sig Generation	CO
Ed25519 Signature Verification	EdDSA Signature Verification over Curve25519	Ed25519 Sig Verification	CO
Ed25519 Key Agreement	Ed25519 Key Agreement	Ed25519 Key Agreement	CO
ECIES	Elliptic Curve encrypt	ECIES Encrypt	CO
ANSI X9.63 Key Derivation	SHA-1 hash-based key derivation function	SHA-1	CO
SP800-56C Key Derivation (HKDF)	SHA-256 hash-based key derivation function	SHA-256	CO
RFC 6637 Key Derivation	SHA hash based key derivation function	SHA-256, SHA-512, AES-128, AES-256	CO
OMAC Message Authentication Code Generation and Verification	One-Key CBC MAC using 128-bit key	OMAC	CO
Message digest generation.	Message digest generation using non-approved algorithms	MD2, MD4, MD5, RIPEMD	CO
Authenticated Encryption / decryption	Encrypt a plaintext / Decrypt a ciphertext	AES-GCM using external IV	CO
(other) symmetric encryption / decryption	symmetric encryption / decryption using non-approved algorithms	Blowfish, CAST5, DES, RC2, RC4	CO

Table 13 - Non-Approved Services

5 Software/Firmware security

5.1 Integrity Techniques

The Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1], which is made up of a single component, is provided in the form of binary executable code. A software integrity test is performed on the runtime image of the module. The HMAC-SHA256 implemented in the module is used as the approved algorithm for the integrity test. If the test fails, the module enters an error state where no cryptographic services are provided, and data output is prohibited i.e., the module is not operational.

5.2 On-Demand Integrity Test

Integrity test is performed as part of the Pre-Operational Self-Tests. It is automatically executed at power-on. Integrity test on demand is performed by power-cycling the computing platform .

6 Operational Environment

The Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1] operates in a modifiable operational environment per FIPS 140-3 level 1 specifications. The module is supplied as part of Device OS, a commercially available general-purpose operating system executing on the computing platforms specified in section 2.

7 Physical Security

The FIPS 140-3 physical security requirements do not apply to the Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1], since it is a software module.

8 Non-invasive Security

Currently, the ISO/IEC 19790:2012 non-invasive security area is not required by FIPS 140-3 (see NIST SP 800-140F). The requirements of this area are not applicable to the module.

9 Sensitive Security Parameter Management

The following table summarizes the keys and Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module:

Key/ SSP Name / Type	Strength	Security Function and Cert. Number	Generation	Import/ Export	Establishment	Storage	Zeroisation	Use and related keys
AES Key / CSP	128 to 256 bits	AES-CBC (A2792, A2794, A2865, A2866) AES-CCM (A2796, A2868) AES-CFB128 (A2793, A2794, A2865, A2866) AES-CFB8 (A27494, A2866) AES-CTR (A2794, A2796, A2866, A2868) AES-ECB (A2793, A2794, A2796, A2865, A2866, A2868) AES-GCM (A2796, A2868) AES-OFB (A2794, A2866) AES-XTS (A2793, A2865)	N/A	Import from calling application No Export	N/A	RAM	Automatic zeroisation when structure is deallocated or when the system is powered down	**Use**: Symmetric Encryption and Decryption **Related keys**: N/A
AES Key-wrapping key / CSP	128 to 256 bits	AES-KW (A2794, A2866)	N/A	Import from calling application No Export	N/A	RAM	Automatic zeroisation when structure is deallocated or when the system is powered down	**Use**: Key Wrapping **Related keys**: N/A
HMAC Key / CSP	128-256 bits	HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, HMAC-SHA-512/256 (A2797, A2869, A2795, A2867, A2798, A2870)	N/A	Import from calling application No Export	N/A	RAM	Automatic zeroisation when structure is deallocated or when the system is powered down	**Use**: Message authentication code generation (HMAC) **Related keys**: N/A
ECDSA public key (including intermediate keygen values) PSP	112 to 256 bits	ECDSA KeyGen (A2797, A2869)	The key pairs are generated conformant to SP800-133r2 (CKG) using FIPS186-4 Key	Import and Export to calling applicat	N/A	RAM	Automatic zeroisation when structure is deallocated or when the system is powered	**Use**: Digital Signature verification **Related keys**: DRBG internal

Key/ SSP Name / Type	Strength	Security Function and Cert. Number	Generation	Import/ Export	Establishment	Storage	Zeroisation	Use and related keys
			Generation method, and the random value used in the key generation is generated using SP800-90ARev1 DRBG	ion. Intermediate keygen values are not output.			down. Intermediate keygen values are zeroized before the module returns from the key generation function.	state, ECDSA private key
ECDSA private key (including intermediate keygen values) CSP								**Use**: Digital Signature generation **Related keys**: DRBG internal state, ECDSA public key
RSA public key / PSP	112 to 150 bits	RSA SigGen, RSA SigVer (A2797, A2869)	N/A	Import from calling application No Export.	N/A	RAM	Automatic zeroisation when structure is deallocated or when the system is powered down.	**Use**: Digital Signature verification **Related keys**: DRBG internal state, RSA private key
RSA private key / CSP								**Use**: Digital Signature generation **Related keys**: DRBG internal state, RSA public key
DRBG Entropy Input / CSP (IG D.L)	256 bits	Random Number Generation E14, E15 (see PUD referenced in section 11.2)	Obtained from two entropy sources	N/A	N/A	RAM	When the system is powered down	**Use** Random Number Generation **Related keys**: DRBG seed
DRBG Seed / CSP (IG D.L)	256 bits	CTR_DRBG (A2797, A2869, A2796, A2868, A2795, A2867, A2794, A2866)	Derived from entropy input string as defined by SP800-90ARev1	N/A	N/A	RAM	When the system is powered down	**Use** Random Number Generation **Related keys**: DRBG entropy input, DRBG internal state
DRBG internal state: V value and Key / CSP (IG D.L)	256 bits	CTR_DRBG (A2797, A2869, A2796, A2868, A2795, A2867, A2794, A2866)	Derived from seed as defined by SP800-90Arev1	N/A	N/A	RAM	When the system is powered down	**Use**: Random Number Generation **Related keys**: DRBG seed

Table 14 - SSPs

9.1 Random Number Generation

A NIST approved deterministic random bit generator based on a block cipher as specified in NIST [SP 800-90ARev1] is used. The DRBG is a CTR_DRBG using AES-256 with derivation function and without prediction resistance. The random numbers used for key generation are all generated by CTR_DRBG in this module. Per section 10.2.1.1 of [SP 800-90ARev1], the internal state of CTR_DRBG is the value V and Key. The module performs DRBG health tests according to section 11.3 of [SP800-90Arev1].

The module also performs DRBG health tests according to section 11.3 of [SP800-90ARev1].

No non-DRBG functions or instances are able to access the DRBG internal state

The deterministic random bit generators are seeded by "*read_random*". The *read_random* is the kernel space interface. Two entropy sources (one non-physical entropy source and one physical entropy source) residing within the TOEPP provide the random bits. The output of entropy pool provides 256-bits of entropy to seed and reseed SP800-90ARev1 DRBG during initialization (seed) and reseeding (reseed).

Name	Minimum number of bits of entropy	Conditioning Components (CAVP number if vetted)
ESV Cert #E14: Apple corecrypto physical entropy source	256 bits	The entropy source consists of twenty-four Free Ring Oscillator (FROs) with a vetted conditioning function SHA-256 (ACVP cert. # C1223)
ESV Cert #E15: Apple corecrypto non-physical entropy source	256 bits	The non-physical entropy source is based upon interrupt timings with a vetted conditioning function SHA-256 (ACVP certs. # A2797, A2869

Table 15 – Entropy Sources

9.2 Key/SSP Generation

The module generates Keys and SSPs in accordance with FIPS 140-3 IG D.H. The cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric (ECDSA) keys as per [SP800-133r2] section 4 example 1 (vendor affirmed), compliant with [FIPS186-4], and using DRBG compliant with [SP800-90ARev1]. A seed (i.e., the random value) used in asymmetric key generation is obtained from [SP800-90ARev1] DRBG. The key generation service for ECDSA as well as the [SP 800-90ARev1] DRBG have been ACVT tested with algorithm certificates found in Table 6.

9.3 Keys/SSPs Establishment

The module provides the following key/SSP establishment services in the Approved mode:

- AES-Key Wrapping: The module implements a Key Transport Scheme (KTS) using AES-KW compliant to [SP800-38F], IG D.G. The SSP establishment methodology provides between 128 and 256 bits of encryption strength.

9.4 Keys/SSPs Import/Export

All keys and SSPs that are entered from, or output to module, are entered from or output to the invoking application running on the same device. Keys/ SSPs entered into the module are electronically entered in plain text form. The module only outputs ECDSA keys in plain text form when key generation service is requested by the calling application.

9.5 Keys/SSPs Storage

Name	Description	Persistence Type
RAM	The module stores ephemeral keys/SSPs in RAM provided by the operational environment. They are received for use or generated by the module only at the command of the calling application. The operating system protects all keys/SSPs through the memory separation and protection mechanisms. No process other than the module itself can access the keys/SSPs in its process' memory.	dynamic

Table 16 - Storage Areas

9.6 Keys/SSPs Zeroization

Keys and SSPs are explicitly zeroised when the appropriate context object is destroyed or when the system is powered down. Input and output interfaces are inhibited while zeroisation is performed.

10 Self-tests

While the module is executing the self-tests, services are not available, and input and output are inhibited. If the test fails either pre-operational and conditional self-tests, the module reports an error message indicating the cause of the failure and enters the Error State (See section 10.3). The module permits operators to initiate the pre-operational and conditional self-tests on demand and periodic testing of the module by rebooting the system (i.e., power-cycling).

10.1 Pre-operational Software Integrity Test

The module performs a pre-operational software integrity test automatically when the module is loaded into memory (i.e., at power on) before the module transitions to the operational state. A software integrity test is performed on the runtime image of the Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1] with HMAC-SHA256 which is an approved integrity technique. Prior to using HMAC-SHA-256, a Conditional Cryptographic Algorithm Self-Tests (CASTs) is performed.

Algorithm	Test Properties	Test Method	Type	Indicator	Details
HMAC-SHA-256	112-bit key	Message Authentication	Software Integrity	Module successful execution	The HMAC value of the runtime image is recalculated and compared with the stored HMAC value pre-computed at compilation time

Table 17 – Pre-Operational Self-Tests

10.2 Conditional Self-Tests

10.2.1 Conditional Cryptographic Algorithm Self-Tests

In addition to the pre-operational software integrity test described in Section 10.1, the module runs the CASTs for all cryptographic functions of each approved cryptographic algorithm implemented by the module each time the module starts.

Algorithm	Test Properties	Test Method	Type	Indicator	Details	Condition
AES-CBC AES-XTS AES-ECB	128-bit key	KAT	CAST	Module becomes operational	Encryption	Test runs at Power-on before the integrity test
AES-CBC AES-ECB	128-bit key	KAT	CAST	Module becomes operational	Decryption	Test runs at Power-on before the integrity test
AES-CCM	128-bit key	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at Power-on before the integrity test
AES-CCM AES-GCM	128-bit key	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at Power-on before the integrity test
CTR_DRBG	AES 128-bit key	KAT	CAST	Module becomes	KAT and Health test per SP800-90Arev1 section	Test runs at Power-on before the integrity

Algorithm	Test Properties	Test Method	Type	Indicator	Details	Condition
				operational	11.3	test
HMAC-SHA256	SHA2-256	KAT	CAST	Module becomes operational	CAST is performed prior to module’s pre-operational software integrity test	Test runs at Power-on before the integrity test
HMAC-SHA-1	SHA-1	KAT	CAST	Module becomes operational	MAC	Test runs at Power-on before the integrity test
HMAC-SHA-512	SHA-512	KAT	CAST	Module becomes operational	MAC	Test runs at Power-on before the integrity test
SHA-1 SHA-256 SHA-512	CAST is covered by higher level HMAC KAT per IG 10.3.B	KAT	CAST	Module becomes operational	Message digest	Test runs at Power-on before the integrity test
RSA Signature Generation	2048-bit modulus with SHA-256	KAT	CAST	Module becomes operational	Sign	Test runs at Power-on before the integrity test
RSA Signature Verification	2048-bit modulus with SHA-256	KAT	CAST	Module becomes operational	Verify	Test runs at Power-on before the integrity test
ECDSA Signature Generation	P-224 curve with SHA-224	KAT	CAST	Module becomes operational	Sign	Test runs at Power-on before the integrity test
ECDSA Signature Verification	P-224 curve with SHA-224	KAT	CAST	Module becomes operational	Verify	Test runs at Power-on before the integrity test

Table 18 - Self-Tests

10.2.2Conditional Pairwise Consistency Test

The Apple corecrypto Module v12.0 [Apple silicon, Kernel, Software, SL1] generates ECDSA asymmetric key pairs and performs a pair-wise consistency tests on the newly generated key pairs.

10.3 Error States

If any of the self-tests described in Sections 10.1, 10.2.1 or 10.2.2 fail, the module reports the cause of the error and enters an error state. In the Error State, no cryptographic services are provided, and data output is prohibited. The only method to recover from the error state is to power cycle the device which results in the module being reloaded into memory and reperforming the pre-operational test and the Conditional algorithm self-tests. The module will only enter into the operational state after successfully passing the pre-operational self-test and the conditional self-tests.

State Name	Description	Conditions	Recovery Method	Indicator
Error State	The HMAC-SHA-256 value computed over the module did not match the pre-computed value	Pre-operational Software Integrity Test failure	module reset	Error message "FAILED: fipspost_post_integrity" is sent to the caller
Error State	The computed value in the invoked Conditional CAST did not match the known value	Conditional CAST failure	module reset	Error message "FAILED:<*event*>" is sent to the caller (<*event*> refers to any of the cryptographic functions listed in Table 18 - Self-Tests.)
Error State	The signature failed to verify successfully in the Conditional PCT.	Conditional PCT failure	module reset	Error message "*CCEC_GENERATE_KEY_CONSISTENCY*" returned for ECDSA Key Generation

Table 19- Error states

11 Life-cycle assurance

11.1 Delivery and Operation

The module is built into DeviceOS defined in section 2 and delivered with Device OS. There is no standalone delivery of the module as a software library.

The vendor's internal development process guarantees that the correct version of module goes with its intended Device OS version. For additional assurance, the module is digitally signed by vendor, and it is verified during the integration into Device OS.

This digital signature-based integrity protection during the delivery/integration process is not to be confused with the HMAC-256 based integrity check performed by the module itself as part of its pre-operational self-tests.

11.2 Administrator Guidance

The Approved mode of operation is configured in the system by default and can only be transitioned into the non-Approved mode by calling one of the non-Approved services listed in Table 13 - Non-Approved Services. If the device starts up successfully, then the module has passed all self-tests and is operating in the Approved mode.

The ESV Public Use Document (PUD) reference for physical entropy source is:
https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/entropy/E14_PublicUse.pdf

The ESV Public Use Document (PUD) reference for non-physical entropy source is:
https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/entropy/E15_PublicUse.pdf

Apple Platform Certifications guide [platform certifications] and Apple Platform Security guide [SEC] are provided by Apple which offers IT System Administrators with the necessary technical information to ensure FIPS 140-3 Compliance of the deployed systems. This guide walks the reader through the system's assertion of cryptographic module integrity and the steps necessary if module integrity requires remediation.

11.3 Non-Administrator Guidance

Not Applicable

11.4 Design and Rules

The Crypto Officer shall consider the following requirements and restrictions when using the module:

- **AES-GCM internal IV** is constructed in compliance with IG C.H scenario 1. The GCM IV generation follows RFC 4106 and shall only be used for the IPsec protocol version 3. When the IV in RFC 4106 exhausts the maximum number of possible values for a given security association, either party to the security association that encounters this condition triggers a rekeying with IKEv2 to establish a new encryption key for the security association. The module uses RFC 7296 compliant IKEv2 to establish the shared secret SKEYSEED from which the AES-GCM encryption keys are derived. In case the module's power is lost and then restored, the key used for the AES GCM encryption/decryption shall be re-distributed. This condition is not enforced by the module.
  This protocol has not been reviewed or tested by the CAVP and CMVP.

- **AES-XTS** mode is only approved for hardware storage applications. The length of the AES-XTS data unit does not exceed 2^{20} blocks. The module checks explicitly that Key_1 ≠ Key_2 before using the keys in the XTS-Algorithm to process data with them compliant with IG C.I.
- **RSA modulus size (IG C.F)**: In compliance with FIPS 186-4, the RSA signature verification is greater or equal to 1024 bits. All supported RSA modulus sizes have been CAVP tested.
- **Legacy use (IG C.M)**: Per SP800-131r2, the SHA-1 within FIPS 186-4 RSA and ECDSA Digital Signature Verification is used in approved mode (for legacy use), the RSA 1024-bit modulus is used in approved mode for FIPS 186-4 signature verification (for legacy use).

11.5 End of Life

The module secure sanitization is accomplished by first powering the module down, which will zeroize all SSPs within volatile memory. Following the power-down, an uninstall by way of system wipe or system update will zeroize the binary file listed in section 2.9.

12 Mitigation of other attacks

The module does not claim mitigation of other attacks.

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
CAVP	Cryptographic Algorithm Validation Program
CAST	Cryptographic Algorithm Self-Test
CAST5	A symmetric-key 64-bit block cipher with 128-bit key
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter Mode
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ESVP	Entropy Source Validation Program
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
KAT	Known Answer Test
KDF	Key Derivation Function
KEXT	Kernel Extension
KW	AES Key Wrap
MAC	Message Authentication Code
KPI	Kernel Programming Interface
NIST	National Institute of Science and Technology
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PKG	Key-Pair Generation
PKV	Public Key Validation
PSS	Probabilistic Signature Scheme
PUD	Public Use Document
RSA	Rivest, Shamir, Addleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
TOEPP	Tested Operational Environment Physical Perimeter
XTS	XEX-based Tweaked-codebook mode with cipher text Stealing

Appendix B. References

FIPS140-3	FIPS PUB 140-3 - Security Requirements for Cryptographic Modules March 2019 https://doi.org/10.6028/NIST.FIPS.140-3
SP 800-140x	CMVP FIPS 140-3 Related Reference https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-standards
FIPS140-3_IG	Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program August 2023 https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-ig-announcements
FIPS140-3_MM	CMVP FIPS 140-3 Draft Management Manual https://csrc.nist.gov/CSRC/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/Draft%20FIPS-140-3-CMVP%20Management%20Manual%2009-18-2020.pdf
SP 800-140	FIPS 140-3 Derived Test Requirements (DTR) https://csrc.nist.gov/publications/detail/sp/800-140/final
SP 800-140A	CMVP Documentation Requirements https://csrc.nist.gov/publications/detail/sp/800-140a/final
SP 800-140B	CMVP Security Policy Requirements https://csrc.nist.gov/publications/detail/sp/800-140b/final
SP 800-140C	CMVP Approved Security Functions https://csrc.nist.gov/publications/detail/sp/800-140c/final
SP 800-140D	CMVP Approved Sensitive Security Parameter Generation and Establishment Methods https://csrc.nist.gov/publications/detail/sp/800-140d/final
SP 800-140E	CMVP Approved Authentication Mechanisms https://csrc.nist.gov/publications/detail/sp/800-140e/final
SP 800-140F	CMVP Approved Non-Invasive Attack Mitigation Test Metrics https://csrc.nist.gov/publications/detail/sp/800-140f/final
FIPS180-4	Secure Hash Standard (SHS) March 2012 http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf
FIPS186-4	Digital Signature Standard (DSS) July 2013 http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf
FIPS197	Advanced Encryption Standard November 2001 http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
FIPS198-1	The Keyed Hash Message Authentication Code (HMAC) July 2008 http://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf
PKCS#1	Public Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1 February 2003 http://www.ietf.org/rfc/rfc3447.txt
RFC3394	Advanced Encryption Standard (AES) Key Wrap Algorithm September 2002 http://www.ietf.org/rfc/rfc3394.txt

RFC5649	Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm September 2009 http://www.ietf.org/rfc/rfc5649.txt
SP800-38A	NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques December 2001 http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf
SP800-38C	NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality May 2004 http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf
SP800-38D	NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC November 2007 http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf
SP800-38E	NIST Special Publication 800-38E - Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices January 2010 http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf
SP800-38F	NIST Special Publication 800-38F - Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping December 2012 http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf
SP800-56Cr2	Recommendation for Key-Derivation Methods in Key-Establishment Schemes August 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Cr2.pdf
SP800-57	NIST Special Publication 800-57 Part 1 Revision 5 - Recommendation for Key Management Part 1: General May 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf
SP800-67	NIST Special Publication 800-67 Revision 1 - Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher January 2012 http://csrc.nist.gov/publications/nistpubs/800-67-Rev1/SP-800-67-Rev1.pdf
SP800-90Ar1	NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators June 2015 http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf
SP800-90B	NIST Special Publication 800-90B - Recommendation for the Entropy Sources Used for Random Bit Generation January 2018 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90B.pdf
SP800-108	NIST Special Publication 800-108r1 - Recommendation for Key Derivation Using Pseudorandom Functions Aug 2022 https://doi.org/10.6028/NIST.SP.800-108r1
SP800-131Ar2	Transitioning the Use of Cryptographic Algorithms and Key Lengths March 2019 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar2.pdf
SP800-133r2	Recommendation for Cryptographic Key Generation June 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-133r2.pdf

SP800-135	NIST Special Publication 800-135 Revision 1 - Recommendation for Existing Application-Specific Key Derivation Functions December 2011 http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-135r1.pdf
SEC	Apple Platform Security https://support.apple.com/guide/security/welcome/web https://manuals.info.apple.com/MANUALS/1000/MA1902/en_US/apple-platform-security-guide.pdf
platform certifications	Apple Platform Certifications https://support.apple.com/guide/certifications/welcome/web